"integration components." Integration components serve as interfaces to existing factory systems and provide capabilities for running APC Plans. An "APC Plan" is an application program called to perform some specific task, as is discussed more fully below. Each of the integration components, e.g., the APC system manager 540, the equipment interface 510, the sensor interface 515 [[520]], the machine interface 535, and the application interface 545, are, in this particular embodiment, software-implemented. They are programmed in C++ using object-oriented programming techniques as are known in the art.

B1

Please replace the paragraph at p. 12, lines 2 – 20 with the following amended paragraph:

In this particular embodiment, the APC System also includes a **P**lan **E**xecution **M**anager 542 ("PEM[[,]]" ~~not shown~~), which is the component primarily responsible for "choreographing" the operation of the APC System 402. This involves interpreting APCFW Plans, executing main scripts and subscripts, and invoking event scripts as events dictate. A variety of plans, scripts, and subscripts may be used in various implementations. For instance, the present embodiment includes, but is not limited to, the following plans:

B2

- data collection plan - a data structure used exclusively by a sensor interface and machine interface defined by the PEM, the requirements in which what data should be collected from a specific processing equipment, and how that data should be reported back to PEM;
- duration plan – defines the trigger conditions and the delays when triggers (*i.e.*, conditions that causes add-on sensors to act upon, *e.g.*, start data collection, stop data collection) happen;
- reporting plan – defines what to do with the collected data, as well as when to signal the data's availability; and
- sampling plan – defines the frequency at which the data is to be collected by the external sensor;

However, the number and function of various plans, scripts, and subscripts will be implementation specific.

Please replace the paragraph at p. 12, line 29 to p. 13, line 2 with the following amended paragraph:

The PEM 542 then delegates responsibility to run the plan to a **P**lan **E**xecutor ("PE") [[542]] (not shown). The PE [[542]] sequentially executes the plan and reports completion of the plan or errors in the execution of the plan to the PEM 542. Thus, while the PEM 542 is responsible for the overall management of all plans executed, a PE 542 is responsible for running only one plan. The PE [[542]] is created by the PEM 542, exists for the life of the plan, and is destroyed by the PEM 542 after reporting that the plan is completed or aborted. Each PE [[542]] executes a main script and zero or more event scripts. A PEM 542 can start multiple plans concurrently via multiple Plan Executors [[542]].

B3

Please replace the paragraph at p. 13, line 19 to p. 14, line 2 with the following amended paragraph:

The sensor interface 515 and the application interface 545 collect data generated by the sensors (not shown) monitoring the operation of the fabrication tool 410. The sensor interface 515 provides the appropriate interface environment to communicate with external sensors such as LabVIEW® or other sensor, bus-based data acquisition software. The application interface 545 provides the appropriate interface environment to execute control plug-in applications such as LabVIEW, Mathematica, ModelWare, MatLab, Simca 4000, and Excel. More particularly, the sensors may be supplied with the fabrication tool 410 by the **o**riginal **e**quipment **m**anfucturer ("OEM") or they may be "add-on" sensors installed subsequent to acquisition from the OEM. The sensor interface 515 collects data generated by add-on sensors and the application interface 545 collects data generated by OEM sensors. The sensors may generate data on, for instance, the pressure and temperature of the operating conditions. The plug-in applications collect the data from the sensors and transmit it to the application interface 545. The machine interface 535, the application interface 545, and the sensor interface 515 use a common set of functionality to collect data to be used. The equipment interface 510 gathers the respective data collected by the application interface 545 and the sensor interface 515. The equipment interface 510 then

B4

transmits the gathered data to the machine interface 535 resident on a workstation 430 ~~over the line 420~~.

Please replace the paragraph at p. 17, line 32 to p. 18, line 11 with the following amended paragraph:

For data collection, the **P**lan **E**xecutor ("PE") 542 sends a **D**ata **C**ollection **Plan** ("DCPlan") to the sensor interface 515 [[520]] or the application interface 545 before data collection. The sensor interface 515 [[520]] and the application interface 545 parse the information in the DCPlan and forward the specified data to the respective sensors (not shown). The specified data may include duration plan, sampling plan, observables, triggers, and limits (*i.e.*, the set-point or value defining the trigger). At this point, the available parameters become known, and the equipment interface 510 may populate the data store 570 accordingly. In addition, the sensor interface 515 [[520]] and application interface 545 are also responsible for forwarding the specified data to be acquired from the process equipment, in a predefined format and in predetermined time interval back to the PE 542. The requirements are all specified in the DCPlan. For instance, the PE 542 may wish to have temperature data from the fabrication tool 410. The PE 542 will issue a directive to the sensor interface 515 [[520]] to configure itself to collect that data at the rate specified by the PE 542, and the PE 542 will instruct the sensor interface 515 [[520]] to report the data back to the PE 542 at a specified reporting rate.

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of the claims: